MVA '92 IAPR Workshop on Machine Vision Applications Dec. 7-9,1992, Tokyo

Representing and Utilising Knowledge for Understanding Structured Documents

T.A. Bayer
Daimler-Benz, Institute for Information Technology

Wilhelm-Runge-Str. 11
7900 Ulm
Germany

Abstract

This paper presents a document analysis system which is capable of extracting the semantics of specific text portions of structured documents. The main component of the system is the knowledge representation scheme — called *Fresco*, **F**rame **Re**presentation of **S**tructured **Do**cuments. It allows the definition of knowledge about document components as well as knowledge about analysis algorithms in a *uniform*, simple, but powerful representation formalism. The specific inference algorithm of *Fresco* is presented which combines the two different knowledge sources — the document model and the algorithmic model. The flexibility of the representation formalism *Fresco* and the properties of the inference algorithm are shown in two different applications, in interpreting amount fields on cheques and in analysing business letters.

1 Introduction

In recent years the importance of document image processing (DIP) and OCR has strongly increased. Although it is obvious that many problems are still unsolved, existing OCR systems work sufficiently in specific applications. The level of abstraction of the output of such a system is a character stream annotated by layout information, like line breaks. OCR systems are not able to extract the meaning of specific text portions. However, if this meaning could be extracted, even partially, many interesting applications could be opened up enabling a highly automated processing of information: for example, if a business letter arrives at a company and the recipient and the sender can be extracted, the letter can automatically be sent to the recipient via the local network and presented on screen and can be stored in a central data base; additionally, the prior correspondence with this sender can be retrieved from the data base automatically, etc.

The analysis system presented in this paper focusses on text portions of structured documents; graphics and image components are only extracted, but not further analysed. The system steps beyond the level of a character sequence as the abstraction level of the resulting symbolic description: Specific logical objects of structured documents are mapped into a representation which reveals their meaning. However, the system is not capable of extracting the semantics of arbitrary text. Rather, those logical objects can be completely interpreted which can be described by simple semantic relationships. Examples of such objects are the recipient and the sender on business letters, reference lists and any kind of objects on forms.

In order to understand text components of structured documents starting from the bitmap, several processing steps are necessary, as described e.g. in [2] and [6]. In general, two different knowledge sources are necessary: algorithms transforming the image data into symbolic primitives and a conceptual model describing the document objects that are to be interpreted. Both components should be closely tied together so that the document model can support the preprocessing. The description of the toolbox is not in the scope of this paper (for details refer to [1]). Rather, the paper focusses on the representation language *Fresco* and the inference algorithm for utilising the knowledge modeled within *Fresco*. Results are obtained in two applications, in reading amount fields and interpreting specific components of business letters.

Recent model based approaches for understanding the logical structure of a document image are described e.g. in [3, 4, 5]. They all include a knowledge base about composition rules of logical objects, presentation rules, like font size, alignment, etc., and relationships between logical structure and layout structure. All these features can be found in the analysis system presented here. Additionally, it includes the image analysis algorithms that are applied in the analysis expectation driven resulting in a higher accuracy of the algorithms.

2 The representation scheme Fresco

Fresco is a semantic-net like language, specifically designed for modeling knowledge about structured documents in a declarative manner. It supports the representation of *layout* concepts and *logical* concepts, like *text-block* as an example of a general layout concept or *date* as a logical concept. Additionally, it enables the modeling of properties of analysis algorithms, which, however, is not in the focus of this paper (cf. [1] for details).

Since by means of *Fresco* knowledge about documents and knowledge about analysis algorithms can be represented in a *uniform* formalism, the image analysis and the

interpretation of a document's components are no longer independent, but closely connected: The hypotheses generated during model expansion guide the image analysis procedures, e.g. what image region is to be investigated or what character type — handwritten, machine printed or both — is expected. Since lexical knowledge and grammars are included in Fresco, the meaning of specific text portions of structured documents can be completely extracted.

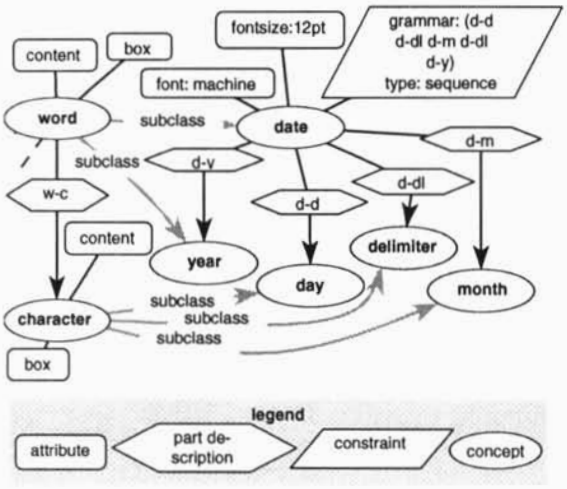

Figure 1: A concept can be described by a set of attributes, by constraints and by structural properties.

The basis of the document representation is the layout model. It defines the general layout concepts, like *text-block*, *line* or *character*, and general layout rules, like *lines-left-aligned*, *words-on-same-baseline*, etc, as explicit concepts of Fresco. Therefore, it is valid for all kinds of documents.

On the other hand, logical concepts describe the knowledge about the application domain, e.g. about business letters or specific forms. Since logical concepts are linked to layout concepts via a *subclass* link, all layout knowledge is inherited to a logical concept. The inherited structures are extended by grammars and by lexical knowledge. Grammars describe the composition rules of concepts in an BNF like syntax, e.g. that a *recipient* consists of *name*, *street*, *city*, etc. Application dependent dictionaries represent the lexical knowledge, e.g. a dictionary of all cities defines the range of strings of the concept *city*. Fig. 1 sketches the properties of a subset of the layout model and of the logical concept *date*.

Since the layout knowledge is valid for any kind of documents, the layout model is a basic component of Fresco. Layout concepts are described on all levels of abstraction, from the very simple level of *connected components* up to the abstract level of a *text block*. In contrast to that, logical concepts only represent objects having a relevant meaning. Thus, logical concepts range from the level of documents down to words and even characters, if a single character itself represents an autonomous object, e.g. a page number. Both concept sets, the layout set and the logical set, are linked together by subclass and by part links. Within this framework concepts of very different classes of structured documents can be modeled. Fig. 2 shows the general layout model along with a small subset of the concepts modeled for the application domain of business letters.

An important feature of *Fresco* is the *reuseability* of the layout concepts, logical concepts and constraint definitions. Once they have been defined in the context of a specific document class, their definition can be used in quite different contexts. E.g. the model of specific lines of an *address block*, *city-line*, *street-line*, etc., can be used both for the definition of the *recipient* and the *sender*. Especially all layout rules, like *words-on-same-baseline*, *left-adjusted*, *centered*, etc., are inherited to logical concepts via the subclass link and need therefore be defined only once.

3 Fresco's Inference Algorithm

The goal of the inference algorithm is to generate instances to layout and logic concept. The inference algorithm is independent from the contents of Fresco; rather, it is only bound to the syntax of this formalism. The different knowledge sources about structured documents are utilised in top-down parsing of the concept structure and bottom-up instantiation: a logical goal concept is specified and the system verifies the complete concept graph defined by the transitive closure of the part relation of the goal concept.

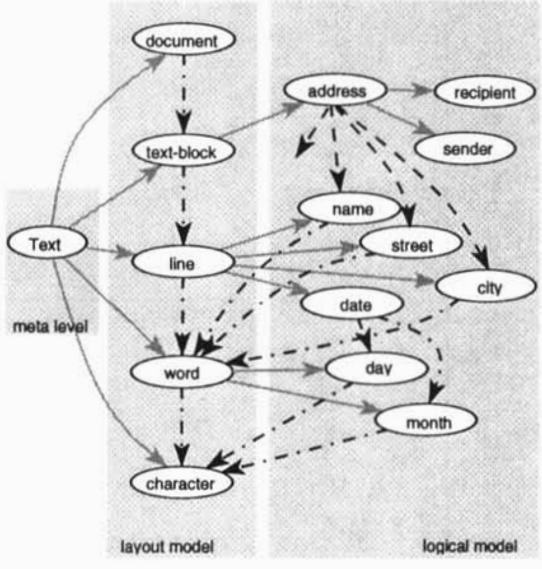

Figure 2: The conceptual graph of general layout model and of a subset of the specific logical model of a business letter.

This process is divided into two interwoven steps, see Fig. 3. The first step — the upper half in Fig. 3 — generates concept hypotheses top-down that have to be verified in the subsequent instantiation phase. The expansion process steps down recursively along the part link until layout concepts are entered. Thus, a set of logical concepts is hypothesised. If, for example, the *date* on a business letter

shall be verified, the set of hypotheses comprises all parts of the date: the digits for the day, month and year and the delimiter information. Since there are different possibilities to write down a date, e.g. the month specification by numbers or by name, the hypothesis set in general contains not only one unique hypothesis list, but a set of different lists.

The second step is invoked when layout concepts are entered during concept expansion of the first step. Its task is to build the layout structure of the document — it performs the document image analysis. Thus, the second step generates instances for layout concepts, like words and lines, which are then passed to the concept graph hypothesised so far. They can be interpreted in the context of the hypothesis set to finally verify the goal concept. During document image analysis algorithms of the toolbox are applied which comprises connectivity analysis, text-graphics separation, text deskewing, character segmentation, line finding, several character classifiers, (specialists for handwritten digits, multifont classifier), etc. (cf. [2] for an overview).

These algorithms are applied *expectation driven*, since the concepts being expanded in the first step provide knowledge about the properties of document objects to be analysed. For example, in most cases the font size can be predicted, the existence of images and graphics on a document can be modeled, and the region of specific text components can be prescribed, like the position of a recipient and the date on a business letter. Having this knowledge, the proper algorithms can be selected from the toolbox and the algorithms employed can be best prepared by adjusting the parameters to the document's model.

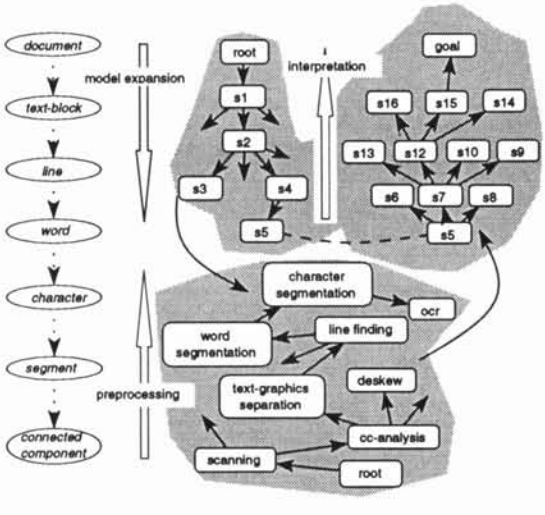

Figure 3: Analysis overview

Each instance of a concept must be evaluated, how well it satisfies the model, since in general the instances computed do not perfectly match the conditions modeled within the concept graph. E.g. if a sequence of lines shall be interpreted as a recipient, it is tested, if they are actually in the region defined by the recipient model. For evaluating instances the fuzzy theory is used. Each attribute and each constraint expression is attached by a fuzzy membership function. Thus, an instance can be evaluated by combining the fuzzy values of its attributes and constraints; the rule of combination is the minimum. Since fuzzy functions are used for evaluation of instances, the analysis is fault tolerant to a certain degree, dependent on the sharpness of the fuzzy membership functions.

Since an inference step allows different possibilities for interpreting primitives, a search space is expanded which must be parsed according to a specific strategy. Since certainties are calculated for each instance, a best first search is performed by the A*-algorithm. This algorithm guarantees that the smallest number of search nodes must be entered during search. The experiments show that the search effort is extremely small in comparison to the theoretical size of the search space, even if a large number of symbolic primitives have been generated by the image analysis.

4 Results

Based on the general layout model and the toolbox, two different applications have been successfully analysed starting from the very beginning of the analysis, i.e. after scanning.

The first application has been the recognition of amount fields on a specific financial form. 1000 samples of 500 different writers have been available. The task has been to extract the amount field from the bitmap and to recognise the amount completely. The digits of the amount are handprinted and are composed in very different styles. Fig. 4 displays a subset of typical amounts which shows the difficulties: touching characters, broken characters, bad quality writing, speckles, etc.

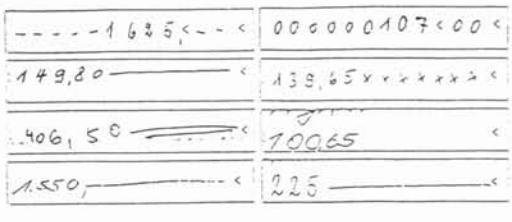

Figure 4: A subset of amount fields showing the variation in print quality, writing style, etc.

During image analysis the layout structure is prepared up to characters. First, the image region is extracted by using the position definition of the form model. This image section is converted into a list of connected components by connectivity analysis. After having removed speckles, the characters are constructed caring especially for broken and touching characters. Before entering the interpretation phase of the analysis, each character is classified by a classifier especially adapated to handwritten digits and few additional characters.

The model of the amount field contains the concepts *dollar*, *cent*, *delimiter* and *digit*. Each concept contains a structural description of its parts, e.g. that the *amount* is composed of at least a *dollar*, followed by the *delimiter* and the *cent* description. The different composition rules

— and there are a lot when investigating the printing styles of 500 writers — are also modeled. The spatial relationships refer mainly to the position of the delimiter between the dollar and the cent, since the delimiter can only be distinguished from the digit "1" by its position, not by its character meaning.

In the second application 23 business letters have been analysed. Among the different logical objects of a business letter the system focussed on the recipient and the date. After having interpreted the recipient and the date, the name, the street, city, P.O. box, the day, the month are available and can be used as an index of a database entry.

Figure 5: Examples of the date and the recipient of the data set of business letters show the variation. The input to the system has been the full document page rather than the image sections displayed here.

The examples were selected from the daily correspondence of different companies with the staff of our institute. The data varies significantly in print quality, font size, position on the document, etc., as Fig. 5 shows. The recipient ranges from a simple three line address to a complex seven line address, including po-box, street, and country. Since many geometrical constraints, lexical constraints and structural possibilities exist, the model of the *recipient* is rather complex, in contrast to the model of the *date*, which can be formulated straight forward.

During image analysis the layout structure is calculated up to the level of characters, if the *date* has been the goal concept, and up to lines if the *recipient* has to be analysed. First, the image is converted into a list of connected regions, which are the basis for skew detection and text/graphics discrimination. After having extracted the text segments the characters are generated in the case of the *date*, otherwise, additional lines and words are generated to extract the recipient. Finally, the characters are classified. Since the model contains a description about the position of both concepts, the image analysis concentrates only on these regions: the expression "(and top left)" describes that the algorithms shall analyse only the upper left section of the document in order to find the *recipient*; "(or (and top right) (and top left))" defines that the *date* lies either in the the top right section or in the top left section. Further model information is used for segmentation and for character classification: if a number of the date shall be classified, a digit classifier is used in conjunction with a general classifier in order to obtain the best possible classification result.

In 21 documents the date could be extracted, whereas the characters of the date could not be classified correctly in two documents. In 19 documents the recipient could be extracted successfully. In the remaining 4 documents many characters have been merged in such a way that the word meaning could not be matched against a dictionary although the dictionary access tolerates up to three character errors in a word.

The search effort for exracting the best interpretation in both applications grows at most qaudratic with the number of primitives extracted by the image analysis algorithms. In most examples the effort is linear to the number of primitives.

5 Conclusion

A document analysis system has been presented which aims at the understanding of text portions of structured documents. It is highly supported by knowledge: It contains a document model covering composition rules, geometric and lexical constraints, and a toolbox of document image analysis algorithms. The results show that restricted text portions can be understood, although it is still far away from understanding arbitrary text and many problems remain unsolved.

One of the urgent problems in modeling documents is the construction of rules defining the general properties of concepts. At the moment, the parameters of these rules, i.e. of the fuzzy functions, are set emipirically. Thus, the next step will be to learn these parameters automatically leading to a set of classifiers adapted to a large set of data. Additionally, the focus of the system is turned to text understanding in future in order to interpret less restricted text portions, like the body of a business letter.

References

[1] Bayer, T.A.: Interpretation of Structured Documents in a Frame System, in: Baird, H.S.: Proceedings of the 4th Workshop on Syntactic and Structural Pattern Recognition, Compton Press, Murray Hill (NJ), 1990, pp. 47-56

[2] Bayer, T.A., Franke, J., Kressel, U., Mandler, E., Oberländer, M.F., Schürmann, J.: Towards the Understanding of Printed Documents, in: Baird, H.S., Bunke, H., Yamamoto, K. (eds.): Structured Document Image Analysis, Springer Verlag, New-York, 1992.

[3] Dengel, A., Bleisinger, R., Hoch, R., Fein, F., Hönes, F.: From Paper to Office Document Standard Representation, in: IEEE Computer, July 1992, pp. 63-67

[4] Ingold, R., Armangil, D.: A Top-down Document Analysis Method for Logical Structure Recognition, in: Proceedings of ICDAR, St. Malo, 1991, pp. 41 -49

[5] Kise, K., Momota, K., Yamaoka, M., Sugiyama, J., Babguchi, N., Tezuka, Y.: Model Based Understanding of Document Images, Proc. of MVA '90, Tokyo, 1990, pp. 471-474

[6] Schürmann, J., Bartneck, N., Bayer, T., Franke. J. Mandler, E., Oberländer, M.: From Pixels to Contents, in: Proceedings of the IEEE, Vol. 80, July 1992, pp. 1101 - 1119